



## Key Data

### Course 8969: Extending Microsoft Dynamics CRM

**Number of Days: Three**

**Format: Instructor-Led**

#### Certification Exams:

This course helps you prepare for the following Microsoft Certified Professional exams:

- 8969 – Extending Microsoft Dynamics™ CRM

#### Certification Track

- Microsoft Business Solutions Certified Master – Microsoft Dynamics CRM Developer

**This course syllabus should be used to determine whether the course is appropriate for the students, based on their current skills and technical training needs.**

**Course content, prices, and availability are subject to change without notice.**

## Course Syllabus

### Extending Microsoft Dynamics CRM

This three-day instructor-led course provides students with the knowledge and skills to develop extensions for Microsoft Dynamics® CRM. The course focuses on extension methods documented in the Microsoft Dynamics CRM SDK.

It also includes content on Microsoft Dynamics CRM Web Service programming, creating and configuring Custom Workflow Activities and Plug-ins, advanced client-side scripting, application integration capabilities and how to create a customer portal solution that connects Microsoft Dynamics CRM to the Internet.

#### Audience

This course is intended for .NET developers who work with Microsoft Dynamics CRM and understand the built-in customization capabilities of the application. Developers attending this course should also understand Web development technologies including client-side programming using DHTML. This course is intended for developers creating extensions for a single Microsoft Dynamics CRM implementation – it does not contain guidance for ISVs creating packaged Microsoft Dynamic CRM add-on products for re-sale.

#### At Course Completion

After completing this course, students will be able to:

- Create applications that use the Microsoft Dynamics CRM Web Services to perform actions on the Microsoft Dynamics CRM platform.
- Create, configure and debug Custom Workflow Activities.
- Create, configure and debug .NET assemblies to use in Plug-ins.
- Create and debug client-side code for Microsoft Dynamic CRM form events.
- Integrate other applications with Microsoft Dynamics CRM through the use of URL addressable forms, IFrames, and custom buttons, menus, and navigation areas added to Microsoft Dynamics CRM.
- Recognize licensing and security issues related to building extensions to Microsoft Dynamics CRM that connect to the Internet.

#### Prerequisites

Before attending this course, students must have:

- Completed Course 8912A Customization and Configuration in Microsoft Dynamics CRM, or have equivalent knowledge of the customization capabilities of Microsoft Dynamics CRM.
- At least three months experience creating .NET applications using Microsoft® Visual Studio®.
- A good understanding of Web development technologies including programming with DHTML.

#### Student Materials

The student kit includes a comprehensive workbook and other necessary materials for this class.

The following software is provided in the student kit:

- Student CD contains Visual Studio projects used in the course.

## Additional Reading

To help you prepare for this class, review the following resources:

- The Microsoft Dynamics CRM SDK.
- Working with Microsoft Dynamics CRM 4.0, Second Edition Microsoft Press.

## Chapter 1: Extensibility Overview

This Chapter provides an overview for the course. It introduces the features available to extend Microsoft Dynamics CRM.

Lessons
<ul style="list-style-type: none"> <li>▪ Microsoft Dynamics CRM Design Focus</li> <li>▪ Extensibility Features</li> <li>▪ Required Skills</li> <li>▪ Resources</li> </ul>

After completing this Chapter, students will be able to:

- Understand how Microsoft Dynamic CRM is designed to be extended.
- Recognize the main extensibility features.
- Recognize the skills they will need to use the extensibility features.
- Recognize resources that will help them learn more.

## Chapter 2: Microsoft Dynamics CRM Architecture

This Chapter describes the Microsoft Dynamics CRM architecture as it applies to practical decisions when planning extensions to Microsoft Dynamics CRM. A strong understanding of the Microsoft Dynamic CRM architecture provides insight that can be used when developing extensions.

Lessons
<ul style="list-style-type: none"> <li>▪ Extensibility Points</li> <li>▪ Microsoft Office Outlook Clients</li> <li>▪ Layers</li> <li>▪ Application Layer</li> <li>▪ Platform Layer</li> <li>▪ Database Layer</li> </ul>
Demonstration: Using the Metadata Browser
<ul style="list-style-type: none"> <li>▪ Open the Metadata Browser.</li> <li>▪ Import the Opportunity Details Custom Entity.</li> <li>▪ Note how Opportunity Details is related to Opportunities.</li> </ul>

After completing this Chapter, students will be able to:

- Recognize where the available extension features exist within Microsoft Dynamics CRM.
- Understand how the Microsoft Dynamics CRM Clients for Microsoft® Office Outlook® interact with Microsoft Dynamics CRM.
- Understand the basic components of Microsoft Dynamics CRM and the functions they perform.
- Understand how Microsoft Dynamics CRM enforces security.
- Understand how Microsoft Dynamics CRM applies business logic.

- Understand how Microsoft Dynamics CRM uses meta-data.
- Understand how Microsoft Dynamics CRM exposes Web Service APIs.
- Understand the functions of the Microsoft Dynamics CRM Platform.
- Understand how Microsoft Dynamics CRM interacts with the SQL Server data store.

### **Chapter 3: Common Platform Operations**

This Chapter explains how to include the Microsoft Dynamics CRM Web Service APIs into development projects and how to use common methods available for all Microsoft Dynamics CRM entities. This Chapter also explains how data types are implemented in Microsoft Dynamics CRM as well as helper code that developers can use to manage Microsoft Dynamics CRM data types. Finally, the process of handling SOAP exceptions from the Microsoft Dynamics CRM Web Services is described.

<b>Lessons</b>
<ul style="list-style-type: none"> <li>▪ CrmDiscoveryService</li> <li>▪ CrmService</li> <li>▪ crmAuthentication Token</li> <li>▪ Entity Information</li> <li>▪ Microsoft Dynamics CRM Data Types</li> <li>▪ Using Type Helpers</li> <li>▪ Using the Create Method</li> <li>▪ Using the Retrieve Method</li> <li>▪ Using the Update Method</li> <li>▪ Using the Delete Method</li> <li>▪ Using the RetrieveMultiple Method</li> <li>▪ Handling SOAP Exceptions</li> </ul>
<b>Lab 3.1: Importing Leads</b>
<ul style="list-style-type: none"> <li>▪ Create a .NET console application that uses the Create Method to import Leads from a .csv file.</li> </ul>
<b>Lab 3.2: Contact Management Application</b>
<ul style="list-style-type: none"> <li>▪ Create a .NET Web application that allows users to view, update and delete Microsoft Dynamics CRM contact records.</li> </ul>

After completing this Chapter, students will be able to:

- Use the common CrmService methods for all Microsoft entities.
- Use Microsoft Dynamics CRM data types and use the type helper provided in the Microsoft Dynamic CRM SDK.
- Handle SOAP Exceptions generated by the CrmService.

### **Chapter 4: Advanced Platform Operations**

This Chapter explains how to query data and perform actions on the Microsoft Dynamic CRM platform using the Execute Method with the appropriate Request and Response classes. It also describes the use of Filtered Views, the DynamicEntity class and methods to work with the Microsoft Dynamic CRM Metadata.

<b>Lessons</b>
<ul style="list-style-type: none"> <li>▪ Querying Data</li> <li>▪ QueryExpression</li> <li>▪ QueryByAttribute</li> <li>▪ Saving Queries</li> <li>▪ Filtered Views</li> <li>▪ Execute Method</li> <li>▪ Requests and Responses</li> <li>▪ Dynamic Entities</li> <li>▪ Using the Metadata Web Service</li> <li>▪ Caching Metadata</li> </ul>
<b>Lab 4.1: Using Query Expression</b>
<ul style="list-style-type: none"> <li>▪ Create a .NET console application to query the Microsoft Dynamics CRM platform using QueryExpression.</li> </ul>
<b>Lab 4.2: Using Filtered Views</b>
<ul style="list-style-type: none"> <li>▪ Create a .NET console application to query the Microsoft Dynamics CRM platform using Filtered Views.</li> </ul>
<b>Lab 4.3 : Using Request and Response</b>
<ul style="list-style-type: none"> <li>▪ Create a .NET console application that uses the Execute Method to reassign accounts evenly amongst users.</li> </ul>

After completing this Chapter, students will be able to:

- Query Microsoft Dynamics CRM using QueryExpression, QueryByAttribute and Filtered Views.
- Use the CrmService.Execute method.
- Use DynamicEntity.
- Access, use and cache Metadata.

## **Chapter 5: Custom Workflow Activities**

This Chapter explains how to create and set up Custom Workflow Activities.

<b>Lessons</b>
<ul style="list-style-type: none"> <li>▪ Configuring Custom Workflow Activities</li> <li>▪ Workflow Architecture</li> <li>▪ Setting Up Custom Workflow Activity Assemblies</li> <li>▪ Creating Custom Workflow Activities</li> <li>▪ Debugging Custom Workflow Activities</li> </ul>
<b>Demonstration: Configuring a Custom Workflow Activity</b>
<ul style="list-style-type: none"> <li>▪ Create a simple workflow rule that uses a Custom Workflow Activity.</li> </ul>
<b>Lab 5.1: Creating a Custom Workflow Activity</b>
<ul style="list-style-type: none"> <li>▪ Create a simple Custom Workflow Activity and register it using the Plug-in Registration Tool so that the activity can be used in a workflow rule.</li> </ul>

After completing this Chapter, students will be able to:

- Write Custom Workflow Activities.
- Create Workflow rules that use Custom Workflow Activities.
- Use the Plug-in Registration Tool.
- Understand Workflow Architecture.

## Chapter 6: Plug-ins:

This Chapter explains how to extend the functionality of Microsoft Dynamics CRM events by writing custom plug-ins.

Lessons
<ul style="list-style-type: none"> <li>▪ Plug-in Model</li> <li>▪ Plug-ins Overview</li> <li>▪ Event Framework</li> <li>▪ Developing Plug-ins</li> <li>▪ Impersonation in Plug-ins</li> <li>▪ Dynamics Entities and Plug-ins</li> <li>▪ Deploying Plug-ins</li> <li>▪ Debugging Plug-ins</li> </ul>
Lab 6.1 : Creating a Plug-in
<ul style="list-style-type: none"> <li>▪ Create a pre-event plug-in and use the Plug-in Registration tool to register it against the event.</li> </ul>

After completing this Chapter, students will be able to:

- Decide when to use Plug-ins.
- Review the Event Framework.
- Develop, de-bug and deploy Plug-ins.
- Use Dynamic Entities within Plug-ins.

## Chapter 7: Application Event Programming:

This Chapter explains how to write client-side code for Microsoft Dynamics CRM Form and field events. It includes the available form and field events, how to work with form and field values, debugging client-side code and several techniques to improve productivity and solve business problems.

Lessons
<ul style="list-style-type: none"> <li>▪ Form and Field Events</li> <li>▪ Overview of Form and Field Events</li> <li>▪ Accessing Microsoft Dynamics CRM Data Fields</li> <li>▪ Setting Event Dependencies</li> <li>▪ Using Best Practices in Writing Client-side code</li> <li>▪ Debugging Client-side code</li> <li>▪ Using DHTML</li> <li>▪ Developing Code with External Files</li> <li>▪ Requesting External Data</li> <li>▪ Accessing Microsoft Dynamics CRM Web Services</li> </ul>
Lab 7.1: Creating Hierarchical Picklists
<ul style="list-style-type: none"> <li>▪ Implement a hierarchical picklist on the phone call entity.</li> <li>▪ Values in the Call Description picklist will be dependent on the option chosen for the Call Type picklist.</li> </ul>
Lab 7.2 : Using XML Request
<ul style="list-style-type: none"> <li>▪ Use the onChange Event of the postal code field to auto-populate the state and city codes based on the value of the postal code.</li> <li>▪ Pass the postal code to a Web page using a query string.</li> <li>▪ Return the XML to update the state and city fields.</li> <li>▪ Use error handling code.</li> </ul>

After completing this Chapter, students should be able to:

- Use Form and Field events.
- Reference Microsoft Dynamics CRM form values.
- Write and debug client-side code in Microsoft Dynamics CRM.
- Request External Data from form and field events.

## **Chapter 8: Application Integration**

This Chapter explains how to add custom buttons, menus, and navigation items in Microsoft Dynamics CRM to integrate other applications. It also explains how to create applications that have the same appearance and behaviors as Microsoft Dynamics CRM. Finally, it describes how IFrames and URL addressable forms are used to integrate Microsoft Dynamics CRM with other Web applications.

<b>Lessons</b>
<ul style="list-style-type: none"> <li>▪ Overview – Customizing the User Interface</li> <li>▪ Customizing SiteMap</li> <li>▪ Using ISV.Config</li> <li>▪ Customizations and the Outlook Client</li> <li>▪ Using IFrames in Entity Forms</li> <li>▪ Using the Microsoft Dynamics CRM Design Guide</li> <li>▪ URL Addressable Forms</li> <li>▪ IFrames Considerations</li> <li>▪ Retrieving Data using Parameters</li> <li>▪ Dynamic IFrame</li> <li>▪ URL Addressable Forms and Views</li> </ul>
<b>Demonstration: Add External Site to Microsoft Dynamics CRM</b>
<ul style="list-style-type: none"> <li>▪ Display the Metadata Browser in a custom area on the Navigation Bar in Microsoft Dynamics CRM.</li> </ul>
<b>Lab 8.1 : Creating Menus and Buttons</b>
<ul style="list-style-type: none"> <li>▪ Add a custom button on the Contact form.</li> <li>▪ The custom button will open a Web site used to set user’s passwords.</li> </ul>
<b>Lab 8.2 : Creating a Task with Default Data</b>
<ul style="list-style-type: none"> <li>▪ Add a button on the Case form.</li> <li>▪ Use this button to open a new Task form with a specific set of default data.</li> <li>▪ Retrieve some of the data from the Case.</li> </ul>

After completing this Chapter, students will be able to:

- Add custom buttons, menus, and navigation items to the Microsoft Dynamics CRM user interface.
- Use IFrames to integrate other applications into Microsoft Dynamics CRM.
- Perform actions on selected records in a Microsoft Dynamics CRM view.
- Set default data in form fields.
- Create applications with the same appearance and behaviors as Microsoft Dynamics CRM.

## Chapter 9: Building ASP .NET Extensions

This Chapter explains how to configure and deploy custom ASP .Net pages which are deployed on the same Web site as Microsoft Dynamics CRM.

<b>Lessons</b>
<ul style="list-style-type: none"><li>▪ Web.config settings within Microsoft Dynamics CRM</li><li>▪ Authentication within Custom ASP .Net applications</li><li>▪ Deploying Custom ASP .Net Applications</li></ul>
<b>Lab 9.1 : Creating and Deploying an ASP .Net Application</b>
<ul style="list-style-type: none"><li>▪ Build and Deploy a Custom ASP .Net application within the Microsoft Dynamics CRM Web site.</li></ul>

After completing this Chapter, students will be able to:

- Build and Deploy ASP .Net Applications on the Microsoft Dynamics CRM Web site.
- Modify the web.config file.
- Understand the authentication and impersonation mechanism within the application.